



Building rock solid *relationships* and optimising technology every time.

Planning an Agile Project with Team Foundation Server 2010

Team Foundation Server (TFS) is Microsoft's software Application Lifecycle Management tool. It encompasses a range of ALM functionality through work item tracking, planning tools and reports, as well as providing other functionality in the areas of Configuration Management and team collaboration.

The next version of TFS will be released early in 2010. This briefing paper covers at a high level some of the key features involved in planning and tracking progress on an agile project using TFS 2010 from the Project Manager's point of view. Future papers will consider TFS 2010's features from the perspective of each of the other members in a typical agile project team.

Project Planning with TFS

One of the central pillars of the TFS ecosystem is the "Work Item Tracking" system, which provides a rich set of requirement, task, defect and test case definition and reporting tools. When creating a new project in TFS, you are able to choose from a number of project templates. Microsoft includes two of these templates out of the box, but there are a number of third party templates also available. Speaking from experience, it can be very simple to set up your own templates as well. The "Agile" template that Microsoft includes with TFS includes a few simple work items, including User Story, Task, Test Case and Defect.

One of the first things an agile project will need to get started is a backlog of requirements, which are initially captured as a set of User Stories. There are a number of options open to you when choosing how to manage your project backlog. Tools that can be used to manage the project backlog include the Team Explorer plug-in for Visual Studio, the Team System Web Access web client for TFS, the project's SharePoint site, or Excel.

The project SharePoint site is something that has undergone a major refit in TFS 2010. The new-look SharePoint site now contains a Project Dashboard, with some key reports available, as well as a list of the User Stories in the Project Backlog. The Dashboard is equivalent to the functionality available in tools such as Rally, and it can be customised to suit each project. If you are using SharePoint Server (i.e. MOSS) rather than the simpler, but free, Windows SharePoint Services, each TFS user also gets the ability to create their own individual "My Dashboard" page. This provides them with customised views of just the information that they are interested in. A typical Project Dashboard is shown in Figure 1 on the next page.



Building rock solid *relationships* and optimising technology every time.

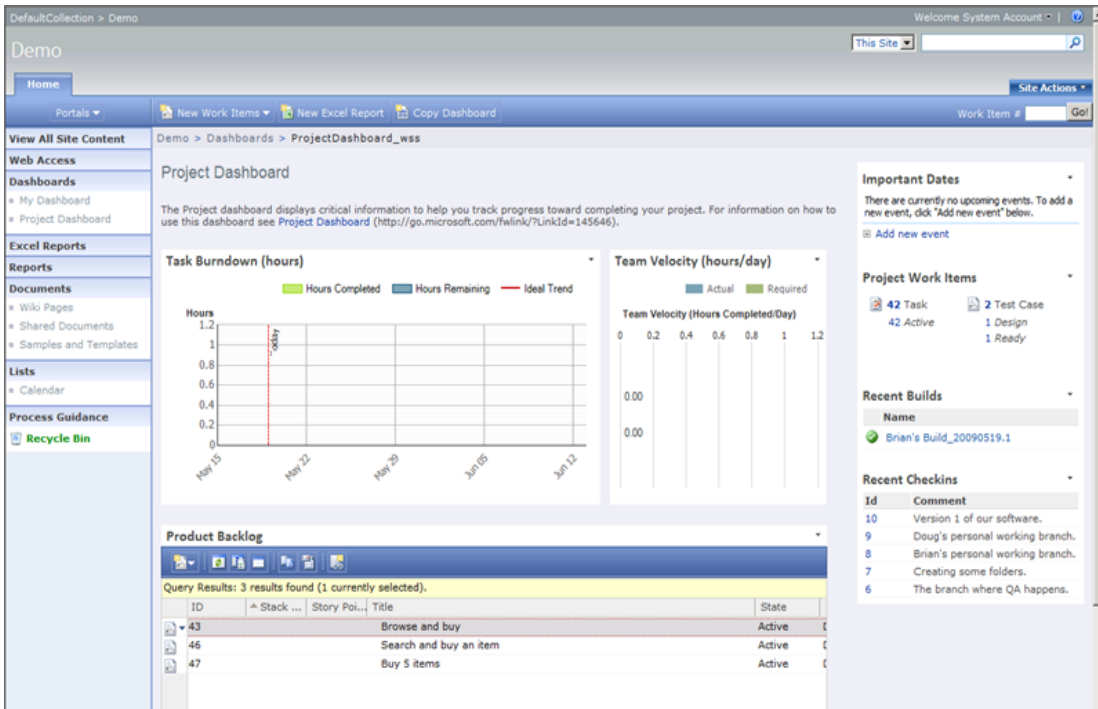


Fig. 1 The Project Dashboard

I mentioned that the Dashboard presents a list of the user stories that make up the project backlog, and this is certainly one tool that can be used effectively to manage the backlog once it's been established. However it is probably not the best tool to use to document user stories in bulk.

The most likely choice of tool for managing TFS work items, especially when performing bulk edits – such as when first capturing the backlog – is Excel. Excel has always been integrated with TFS, but with TFS 2010 Microsoft provides Excel workbook templates for a number of usage scenarios with TFS. These are available from the project SharePoint site. When opened from there, they are configured to automatically access the work item store for the project in question so there is no need to manually set up a connection.

Most notably, there are workbooks supporting the management of the project backlog and for iteration planning. By opening the project backlog workbook, you are able to use the familiar Excel interface to add, remove, or update the user stories quickly and easily. It is also possible to define Iterations in TFS for the project, as well as “Areas”, which are used in TFS to partition, or classify, the work on a project. For example, you might have areas for “User Account Management”, “Shopping Cart”, and “Product Catalogue” in a project that is building an online shopping website.

With the project backlog entered, the next major task will be to plan the first iteration. For this, you can again use Excel, in this case through the Iteration Planning workbook. This workbook contains a set of instructions, and helps a project team to select the items from the



Building rock solid *relationships* and optimising technology every time.

project backlog that will be included in this iteration, establishing the Iteration Backlog. Obviously this is one workbook that will be used again and again, so it makes sense to create a separate copy for each new iteration.

Work Item Type	Status	Priority	Date	Description	Acceptance Criteria	Severity	Iteration	Assignee	
Story	Not Started	3		The system presents each page based on a consistent template.	Each page is based upon a template page, and shares a common set of visual style and user interaction elements with the other pages.	3	Low	Iteration 0	Mike Thompson and Andre Arlino
Story	Not Started	2		Users can browse public sections of the site without logging in.	Acceptance Criteria: Each page is consistent in style and layout. Users can browse public sections of the site without logging in. Default page and much of the content will be public.	3	Low	Iteration 0	Mike Thompson and Andre Arlino
Story	Not Started	2		Users can browse public sections of the site without logging in.	Acceptance criteria is that a user can browse non-secure pages without having to log in, and that secure pages are protected, so that the user is prompted to log in if they decide to use.	3	Low	Iteration 0	Mike Thompson and Andre Arlino
Story	Not Started	3		All users can view the site home page.	Acceptance Criteria: Users are able to view the home page, when logged out, as well as when logged in.	3	Low	Iteration 0	Mike Thompson and Andre Arlino

Fig. 2 The Project Backlog workbook.

Final word

This is a fairly high level overview of the planning process, but it should provide you with some feel for the tools available in the next release of TFS. Other papers in this series will cover the tools available for tracking progress, including the reports available, and how TFS 2010 makes it easy for developers to record their progress towards completing the tasks they've been assigned.

For more information, Microsoft's Brian Harry has a good overview of the Project Management improvements in TFS 2010 at <http://blogs.msdn.com/bharry/archive/2009/05/19/tfs-2010-project-management.aspx>. I suggest you take the time to read this, since he covers other areas of interest that I don't go into here, such as MS Project integration. Also, I have my own blog, <http://nzdotnetdude.wordpress.com/>, where I've been posting more detailed information on the practical use of TFS 2010 from my own experiences.

About the author:

Mark Lawrence is a software architect at Optimization. He has worked in the IT industry for 16 years, including some 10 years working with the .Net framework. Mark is Optimization's Team Foundation Server expert, having used it extensively for more than five years, with both national and multi-national organisations.

Mark can be contacted at mark.lawrence@optimization.co.nz.